



PATENT ABSTRACTS OF JAPAN

(11) Publication number: **06046243 A**(43) Date of publication of application: **18.02.94**

(51) Int. Cl. **H04N 1/387**
G06F 15/62
G06F 15/62
G06F 15/66
H04N 1/41

(21) Application number: **05033814**(22) Date of filing: **24.02.93**(30) Priority: **23.03.92 US 92 857037**(71) Applicant: **RICOH CO LTD**(72) Inventor: **ROBAATO EFU MIRAA**
SUTEIIBUN EMU BURONSUTAIN

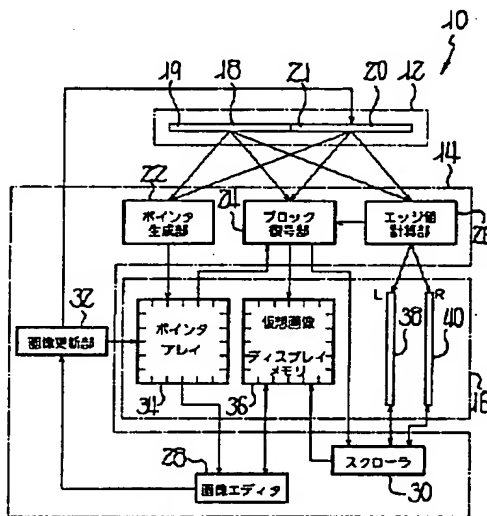
(54) **VIRTUAL EDIT METHOD FOR COMPRESSED**
PICTURE AND ITS DEVICE

(57) Abstract:

PURPOSE: To make the control of a picture possible while being compressed in a picture element level.

CONSTITUTION: In the picture compression system employing the typical picture compression system, a pointer array 34 is provided to point out each of lots of minimum coding units in a compressed picture file. Partial sets of a block is selected as a virtual picture from all blocks of the picture. The virtual picture is edited and each edit block is compressed in an edit block. The edit block is compressed in the edited minimum coding unit and arranged in the edit block area and a pointer to the original minimum coding unit is revised to point out a new minimum coding unit. According to the method, it is possible to revise the pointer array 34 to execute cancellation.

COPYRIGHT: (C)1994,JPO&Japio



(19)日本国特許庁(JP)

(12)公開特許公報(A)

(11)特許出願公開番号

特開平6-46243

(43)公開日 平成6年(1994)2月18日

(51)Int.Cl.⁵

H 0 4 N 1/387

G 0 6 F 15/62

15/66

H 0 4 N 1/41

識別記号

庁内整理番号

F I

技術表示箇所

3 2 0 A 9365-5L

3 3 0 J 8420-5L

B 9070-5C

審査請求 未請求 請求項の数14(全 10 頁)

(21)出願番号 特願平5-33814

(22)出願日 平成5年(1993)2月24日

(31)優先権主張番号 07/857, 037

(32)優先日 1992年3月23日

(33)優先権主張国 米国(U S)

(71)出願人 000006747

株式会社リコー

東京都大田区中馬込1丁目3番6号

(72)発明者 ロバート エフ ミラー

アメリカ合衆国 カリフォルニア州 モデ

スト メスロブ コート 3925

(72)発明者 スティーブン エム ブロンスタイン

アメリカ合衆国 カリフォルニア州 サン

ジョーズ ブリアリーフ サークル

1214

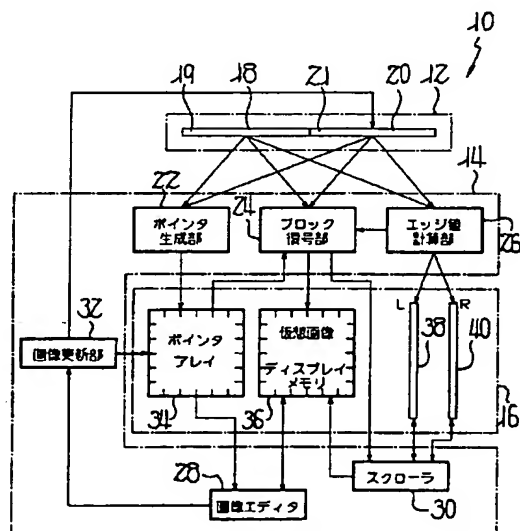
(74)代理人 弁理士 柏木 明 (外1名)

(54)【発明の名称】 圧縮画像の仮想編集方法及びその装置

(57)【要約】

【目的】 画素レベルで圧縮されたままの画像操作を可能とすること。

【構成】 典型的な画像圧縮方式を用いる画像圧縮システムにおいて、ポインタ・アレイ34が圧縮画像ファイル内の多くの最小符号化単位の各々を指し示すように提供される。画像の全てのブロックからブロックの部分集合が仮想画像として選択される。仮想画像は編集され、各々の編集されたブロックは編集ブロック内に圧縮される。編集ブロックは編集された最小符号化単位内に圧縮され編集ブロック領域内に配置されて、原本の最小符号化単位へのポインタは新規の最小符号化単位を指すように変更される。この方法によって、ポインタ・アレイ34は取消し動作を実行するように改変することが可能である。



【特許請求の範囲】

【請求項1】 編集するために圧縮された画像の一部を取扱うための方法であって、前記画像は各々がデジタルデータによって表現されるブロックに分割され、また、各々のブロックでの前記デジタルデータは最小限に符号化した単位に圧縮されており、複数の前記最小限に符号化した単位が圧縮された画像データのペケットを構成し、

ポインタ・アレイを生成し、前記ポインタ・アレイ内の各々のポインタは前記圧縮された画像データのペケット内部の最小限に符号化した単位を示すようになす段階と、前記画像から仮想画像を定義するためにブロックを選択する段階と、前記仮想画像に対応する一組の最小限に符号化した単位を展開する段階と、前記仮想画像を編集して少なくとも一つの編集したブロックを形成する段階と、前記編集したブロックを編集された最小限に符号化した単位に圧縮する段階と、前記編集された最小限に符号化した単位を編集されたブロック・メモリ内に保存する段階と、前記ポインタ・アレイ内のポインタを変更し、前記ポインタが前記編集したブロックに対応し、前記編集されたブロック・メモリ内の前記編集された最小限に符号化した単位を示すように変更する段階と、それにより前記ブロックは、これの対応する最小限に符号化した単位が前記変更する段階において変更されたように前記ポインタ・アレイのポインタによって示され、全体として編集された画像を定義する段階とよりなることを特徴とする圧縮画像の仮想編集方法。

【請求項2】 ブロックを選択する段階が、画像の全体より少ないブロックを選択することを特徴とする請求項1記載の圧縮画像の仮想編集方法。

【請求項3】 編集段階は、ディスプレイ上の仮想画像を観察する人間の利用者との対話形式でコンピュータによって完成されることを特徴とする請求項1記載の圧縮画像の仮想編集方法。

【請求項4】 編集段階、少なくとも一つの編集されたブロックを圧縮する段階及びポインタを変更する段階が1回以上反復されることを特徴とする請求項1記載の圧縮画像の仮想編集方法。

【請求項5】 ブロックが単一の画素よりなることを特徴とする請求項1記載の圧縮画像の仮想編集方法。

【請求項6】 各々のポインタが編集フラグとアドレス値よりなることを特徴とする請求項1記載の圧縮画像の仮想編集方法。

【請求項7】 最小限に符号化した単位は、最小限に符号化した単位のヘッダと最小限に符号化した単位の本体とよりなり、前記最小限に符号化した単位のヘッダは画像ブロック識別子と前記最小限に符号化した単位におけるビット数を表わす長さフィールドとよりなり、また前記最小限に符号化した単位の本体は画像ブロックの品質についての少なくとも一つの相対値と複数の絶対値とよ

りなることを特徴とする請求項1記載の圧縮画像の仮想編集方法。

【請求項8】 最小限に符号化した単位は、ブロックの品質についての差分で符号化した値を含み、さらに、前記仮想画像の辺縁に隣接する複数のブロックの前記品質についての絶対値を決定する段階と、前記絶対値を保持するためのエッジ・テーブルを生成する段階と、前記エッジ・テーブルを使用して展開され前記仮想画像に追加された新規のブロックについての前記品質の絶対値を決定する段階とを有することを特徴とする請求項1記載の圧縮画像の仮想編集方法。

【請求項9】 新規のブロックが、スクロール動作において追加されることを特徴とする請求項8記載の圧縮画像の仮想編集方法。

【請求項10】 編集するために圧縮された画像の一部を取扱うための方法であって、前記画像は各々が少なくとも一つの相対値を含むデジタルデータによって表現されるブロックに分割され、他のブロックの相対値を参照することによりブロックの絶対値が決定されるようになしてあり、

前記圧縮画像の部分に対応する仮想画像を定義するために前記画像からのブロックを選択する段階と、前記仮想画像の辺縁に隣接する複数のブロックの各々について絶対値を決定する段階と、前記絶対値を保持するためのエッジ・テーブルを生成する段階と、前記エッジ・テーブルを使用して展開され前記仮想画像に追加されたブロックについての絶対値を決定する段階とよりなることを特徴とする圧縮画像の仮想編集方法。

【請求項11】 編集するために圧縮画像の一部を取扱うための装置であって、前記画像はブロックに分割され、各々のブロックは各々のブロック内の各々の画素に対応する点でデジタル化された画像を表わす画素データよりなり、また前記画像は圧縮された画像データのペケット内に最小限に符号化した単位として保存され、前記圧縮された画像データペケットにおいて各々の最小限に符号化した単位は一つのブロックに対応しており、各々のポインタが最小限に符号化した単位を示し全体として前記圧縮された画像データを示すポインタ・アレイと、前記圧縮された画像データペケットを保持するための圧縮データメモリと、仮想画像に対応するブロックを保持するための仮想画像メモリと、前記仮想画像の編集されたブロックに対応する編集された最小限に符号化した単位を保持するための編集データメモリと、編集されたブロックに対応する前記ポインタ・アレイ中のポインタを変更するための手段であって、前記編集されたブロックに対応する編集された最小限に符号化した単位を指すように前記ポインタを変更するための手段とよりなることを特徴とする圧縮画像の仮想編集装置。

【請求項12】 圧縮されたデータがJPEG規格に従って圧縮されていることを特徴とする請求項11記載

の圧縮画像の仮想編集装置。

【請求項13】 画素データがグレースケールの濃度を表わすことを特徴とする請求項11項記載の圧縮画像の仮想編集装置。

【請求項14】 画素データは一つ以上のデジタル値よりなり、各々のデジタル値が一つの色の濃度を表現するようになしてあることを特徴とする請求項11項記載の圧縮画像の仮想編集装置。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、圧縮画像の編集分野に関し、より特定すれば、画像全体を展開することなく圧縮画像の部分編集するための圧縮画像の仮想編集方法その装置に関する。

【0002】

【従来の技術】カメラ及びスキャナにとって、現在のカラー画像は、入力した又は操作した画像を表現するデータを含む画像ファイルを保存するために、大量のメモリを要求する。カラー複写機で8.5×11インチ(216×297mm)の用紙、解像度400ドット/インチ(15.75ドット/mm)、各ドットの濃度を表現するために3バイト(24ビット)を用いるような典型的な画像では、44.88メガバイトのメモリを占有することになる。メモリの容量に加え、メモリはリアルタイムの用途、例えば、毎分15枚の速度が一般的な要求とされるカラー複写機などのための広い帯域幅を有する必要がある。これは、帯域幅又はデータ速度で11.22メガバイト/秒を意味する。これら双方の要求が画像システムのコストを押し上げることになる。圧縮技術は、当面の帯域幅と保存容量の問題の両方を解決するために用いられてきた。圧縮によると、画像ファイルは少ないメモリに保存可能であり、画像を表現するのに使用されているビット数が少ないために、何らかの帯域幅の制限されているチャンネル間で高速に移動可能である。圧縮は元の画像データをコンプレッサ回路に通すことで行なわれ、コンプレッサ回路が元のデータに含まれるパターンを統合又は分析して、対応する展開なしには元の画像が容易に識別し得ないような圧縮画像ファイルを生成する。

【0003】画像に編集、伸縮、回転又はその他の処理を加える場合、何らかの又は全ての画素への無作為なアクセスが要求される。処理が完了すれば、編集結果を画像ファイルに保存する必要がある。処理している画像ファイルが圧縮されている場合、従来の無作為な画素へのアクセスを得るための手段は画像をフレーム記憶装置内に展開し、処理を実行し、しかる後、画像を新しい画像ファイルに再圧縮する。これに伴う問題は、展開によりメモリが保存されないことで、フレーム記憶装置にメモリが必要とされるためである。画像全体の圧縮と展開に時間がかかることから帯域幅も低下する。画像システム

において最も高価な表示装置でさえも完全な解像度で画像全体を表示し得ないことから、このさらなる処理時間とメモリはしばしば無駄となる。

【0004】一つの解決策は、制約されたインライン・リアルタイム操作だけを許容し、画像がチャネルを通して一つの圧縮画像から別の画像へ移動するように画像を操作することである。残念なことに、同時に処理できるのは小さな領域だけであり、こうした方式は対話型編集には対応していない。

10 【0005】画像データを圧縮するための圧縮法は周知である。こうした圧縮規格の一つがJ P E G (Joint Photographic Experts Group) である。J P E Gでは、一つの例を示すと、画像は画像の要素又は画素の2次元アレイによって表現されている。画像がグレースケールの場合、各々の画素は濃度値により表現され、また、画像がカラーの場合、画素は幾つかの値によって表現される。画像データを圧縮する以前に、画像は各々が8×8画素のブロックに分割される。各ブロック内で64画素がディスクリート・コサイン変換(DCT変換)を用いて、直流(DC)値と63の交流(AC)値からなる64の周波数振幅に変換される。DC値を直接符号化する代りに、現在のブロックに先行するブロックのDC値を現在のブロックのDC値から減算し、得られた差分DC値を保存する。圧縮する標準的画像は各々の8×8画素ブロック内部で定常的な色と比較的緩やかに変化する平均色相値を有しているので、この方式は大幅に圧縮した画像が得られる。これ以外の大半の画像圧縮方式も、同様の圧縮方式を使用し、また、差分値を用いている。

30 【0006】画像は画素位置の数、通常は、2次元格子で色又は濃度を表わす元の画素データからなる。画素はブロックにまとめられ、各々のブロックが別個に圧縮される。圧縮の過程は、ブロックに関する64の値の一つの差分DC値と63のAC値に変換することと、エントロピー符号化により変換した値を記述するために必要とされるビット数を減少することから構成される。各々のブロックは、最小符号化単位(MCU)に圧縮され、圧縮画像を構成するMCUは識別情報、例えば、画像内のブロック位置、何個のビットがMCU内に含まれているかを表わす長さデータ、及び同様な情報が添付され、MCUは画像ファイル内に順次保存される。長さデータはMCUが長さ変化するため必要とされる。長さはエントロピー符号化の段階で決定される。良く配列されているブロック、例えば、均一な色相のブロックなどは「低いエントロピー」を有すると言われ、高いエントロピーを伴うブロックと比較して、相対的に小さなMCUに圧縮される。エントロピー符号化の過程は、一般に、「符号化」と称しており、それに対して圧縮過程はブロック化、コサイン変換、及び画像データ符号化を含む。明らかなことは、完全な展開が必要ではないため、差分DC

値を復号されただけのMCUから読取ることができる。処理資源におけるこの節約は、コサイン変換及び逆変換が極めて計算を多用する操作であり得ることから重要であろう。

【0007】画像を展開するには、画像ファイルからの各々のMCUは、DC値の差分符号化の逆転操作により、また、逆コサイン変換によりブロックに展開される。ブロック間で緩やかに変化する値の差分符号化は圧縮を増加させるが、画像の一部の展開の困難さも増大させる。これは、部分的画像の幾つかのMCUの展開が部分的画像の一部をなさないMCUの参照を必要とするためである。部分的画像が編集され、再圧縮される場合には、さらなる困難が部分的展開で発生する。圧縮度は画像データに依存しているため、編集された部分的画像は原本画像と比較して圧縮時に異なる容量をなす。その結果、単純に部分的画像からのMCUを編集した部分的画像からのMCUで置換することは画像全体が展開、編集、再圧縮されない限り非実用的である。上記のことから圧縮画像を操作し編集するための方法の改良が必要とされていることは明白である。

【0008】

【発明の目的】本発明は、画素レベルで圧縮された形状のままの画像の操作を可能にする。典型的な画像圧縮方式を使用する画像圧縮システムにおいて、ポインタ・アレイが圧縮画像ファイル中の多くのMCU（最小符号化単位）の各々を指すために提供される。画像のブロック全体からブロックの部分集合が仮想画像として選択される。

【0009】仮想画像が編集され、各々の編集ブロックが編集ブロック内に圧縮され、原本のMCUへのポインタが編集されたMCUを指すように変更され、これが編集画像データ領域内に保存される。この方法では、ポインタ・アレイは取消操作を実行するように変更することが可能である。

【0010】本発明は、全てのMCUが復号されていないようなMCU内の差分符号化した値を取扱うための手段を提供している。エッジ・テーブルは値を保持するために用意されており、ここで、各々の値は仮想画像の辺縁上のブロックについての差分値と結合された場合、仮想画像の辺縁より遠方のブロックを参照することなくそのブロックについての絶対値を提供する。エッジ・テーブル内のエントリーは、ブロックの完全な展開なしに圧縮されたMCUから決定される。さらなるテーブルのために占有される記憶空間より速度が有意義であるような場合に、差分力絶対値を計算する速度を向上させるために、一つ以上のエッジ・テーブルが用意される。絶対値計算は平均してエッジ・テーブルが多いほど高速であるが、これは計算速度がエッジ・テーブルを有する最も近い辺縁までのブロックからの距離に比例するためである。

【0011】本発明は、また、全てのMCUを並べ替え原本の画像データ領域からのMCUを編集した画像データ領域からのMCUで置換するために圧縮された編集画像の再インデックスを取るための手段も提供する。本発明で解説する好適実施例は、JPEG規格に従って圧縮された画像データで動作するが、本発明は、ブロックが可変容量MCUに圧縮されるような、又はデータが差分値で符号化されることによりMCUを完全に復号するためには、別のMCUを参照しなければならないようなあらゆる画像圧縮方式について応用が可能である。

【0012】本発明による画像エディタにおいて、圧縮画像は一つのメモリ領域内に読込まれ、また、編集すべき選択されたブロックが一つのメモリ領域から展開されてエディタに渡され、編集され、コンプレッサ回路によって再圧縮される。再圧縮され、編集されたブロック発議に編集されたブロックメモリ内に配置される。ポインタ・アレイは、アレイ内のポインタが編集後の画像の各々のブロックを指し示すように、編集されたブロック・メモリの一つのメモリ領域内の何れかのブロックを示すために提供される。

【0013】

【実施例】本発明の一実施例を図面に基づいて説明する。まず、図1は本発明による編集システム10の好適実施例を機能的に図示したものである。編集システム10は、圧縮画像メモリ12、画像処理部14及び画像メモリ16を含む。前記圧縮画像メモリ12は、圧縮画像19を含む原本の画像データ領域18と編集したMCU21を保持するための編集画像データ領域20を含む。前記画像処理部14は、ポインタ生成部22、ブロック復号部24、エッジ値計算部26、画像エディタ28、スクローラ30及び画像更新部32を含む。前記画像処理部14のこれらの構成部材は、別個のハードウェア、処理装置により実行されるソフトウェアのサブルーチン群、又は両者の組合せとして実現することが可能である。

【0014】前記画像メモリ16は、ポインタ・アレイ34、ディスプレイ・メモリ36、左エッジ・テーブル38及び右エッジ・テーブル40を含む。前記圧縮画像メモリ12は、ランダム・アクセス・メモリ（RAM）、磁気ディスク又はその他の記憶媒体などのデジタルメモリである。

【0015】前記圧縮画像メモリ12に何らかの編集を行なうまで、前記編集画像データ領域20は空で編集されたMCUを全く含まない。この編集画像データ領域20に割当てられるメモリ量は、実行すべき編集の量に基づく利用者の必要性和多重取消し動作の必要性に従って変化し得る。

【0016】前記画像メモリ16の初期化についてこれから解説する。画像メモリ16は、編集作業の開始時又は圧縮画像メモリ12内に新規に圧縮画像が読込まれた

場合に初期化される。画像はブロックの2次元アレイよりなる。カラー画像において、画像は、第1に複数のカラー層に細分され、各々のカラー層は単一の画像として扱われる。以下の議論では、単一のカラー層画像についてのみ解説するが、議論する原理の範囲は、直接的な方法で複数カラー層に拡張し得るものである。ブロックは、通常8×8画素の方形を含み、非圧縮時には64の濃度値を提供しており、また、画像はこうしたブロックのN×Mアレイにより構成される。各々のブロックが8×8画素であるため、8. 5×11インチでインチ当たり400画素の場合、Nは425、Mは550となり、全体として画像当たり233, 750ブロックとなる。この例について続けると、圧縮画像19は圧縮データを含む233, 750MCUを含むことになる。圧縮画像メモリ12内の画像が編集されて再インデックスが取られなかった場合、編集画像データ領域20は圧縮データのMCUを含み、編集画像データ領域20内の各々のMCUは原本の画像データ領域18内のMCUを置換し、また、ブロックが1回以上編集された場合、領域20は領域18のMCUを置換するための一つ以上のブロックを含むことになるが、実際に領域18内のブロックを置換するのは、一つの編集されたMCUだけである。ポインタ生成部22は領域18を走査してポインタ・アレイ34内に各々のMCUの開始部分へのポインタを配置する。ブロックとポインタの一対一対応により、ポインタ・アレイ34はN×M又は233, 750個のエントリーを含む。

【0017】図2及び図3は、前記ポインタ・アレイ34内のポインタがどのように構成されるかを良く図示している。第1のポインタ35は、圧縮画像19の第1のMCUを示す。他のポインタも同様に、ブロックとそれらに対応するMCUに付随している。ポインタ・アレイ34は、図1に図示したように2次元アレイ内に論理的に配置して画像内のブロックの構成に対応でき、又は、図3に図示したように、1次元アレイ内に構成することが可能である。各々のポインタは領域18内のMCUを示すか又はポインタn+3の場合のように領域20内のMCUを示す。

【0018】図1をさらに参照すると、領域18内の各々のMCUを走査してしまうと、ポインタ生成部22は編集画像データ領域20を走査する。領域20内の各々のMCUは別のMCU内のデータを置換する画像データを含む。置換されたMCUは、MCUが原本のMCUの第1の編集物である場合には、領域18内のMCUを置換するか、又はMCUが既に編集されている場合、領域20内のMCUを置換する。領域20内の各々のMCUは置換されるMCUへのポインタを含む。従って、ポインタ生成部22は領域20内の各々のMCUを単純に走査し、そのMCUに関連するブロックを認識してポインタ・アレイ34内のそのブロックへのポインタを更新す

る。ポインタ生成部22は古い編集から最新の編集へと領域20内を走査し、単一ブロックについて複数の編集が領域20内に存在する場合、そのブロックについて最新のMCUだけがポインタ・アレイ34内のポインタによって示される。

【0019】ポインタ・アレイ34が初期化されると、ブロック復号部24はこのポインタ・アレイ34内のポインタを用いて領域18内の選択されたMCUを検索するが、別の実施例において、ブロック復号部24はポインタ・アレイ34の参照なしに領域18内のMCUを特定している。しかし、処理の重複を回避することからポインタ・アレイ34を用いることが望ましい。ブロック復号部24は圧縮画像メモリ12内に保存された画像全体を復号するが、本発明の幾つかの利点は失われることになる。

【0020】通常、画像の編集は、全画像のわずかな部分にのみ関連する。この部分は、仮想画像が復号された後、エディタ28により完全な画像として操作される。図2は仮想画像44が全体画像42にどのように関連するかを図示している。図2において、全体画像42はN×Mブロックの広がりを持つが、仮想画像44は4×4ブロックの大きさである。

【0021】図1に戻って上記の例を続けると、仮想画像44は編集用に選択された画像であり、ブロック復号部24は仮想画像44の16ブロックだけを復号し得られたブロックをディスプレイ・メモリ36内に配置する。ディスプレイ・メモリ36から仮想画像44が処理され、表示され、又はそのまま保存され得る。重要なことは、画像処理部14は全体画像の全てを展開するための資源と時間の処理に専念する必要がなく、仮想画像44内部のブロックに関連するMCUにだけ集中すれば良い点である。

【0022】初期化の間、ブロック復号部24はエッジ値計算部26と並行して動作している。他の実施例では、これらの動作は逐次的又は全く独立である。エッジ値計算部26は圧縮画像メモリ12内のMCUを走査し、差分であって基準MCUへの参照を要求する各々のMCUの差分値だけを評価する。一つの規格であるJPEG規格では、各々のMCUはDC濃度についての差分値を含み、あらゆるMCUについての基準MCUは復号されるブロックの左にあるブロックについてのMCUである。例えば、ブロックn+2についての絶対DC値を検索するには(図2参照)、差分値がMCUn+2から抽出されてから抽出した値からブロックn+1についての絶対DC値が減算される。しかし、ブロックn+1は仮想画像44内に含まれないため展開されていないので、エッジ値計算部26によりDC値だけがそのブロックから抽出される。全体画像に渡り仮想画像を移動させる際の遅延を回避するため、左右のエッジ・テーブル38, 40が用意されている。

【0023】図4は、エッジ・テーブル38、40とディスプレイ・メモリ36内に保存されている仮想画像44の関連性を良く図示している。仮想画像44は当初にブロックA、B、C、Dにより仕切られている。左エッジ・テーブル38は全体画像42の各行について一つのエントリーを含み、またテーブルはしばしば仮想画像44の境界を越えて到達することになる。各行についてのエントリーはその行の中のブロックについてと仮想画像44内の最も左の桁のブロックの左側までの絶対DC値を含む。例えば、左エッジ・テーブル38のエントリーYはブロックDの左側に存在する全体画像中のブロックについてのDC絶対値である。同様に、右エッジ・テーブル40は仮想画像44の最も右側の桁にあるブロックのDC絶対値についてのエントリーを含む。よって、エッジ・テーブル38、40の各々がM個のエントリーを含むことになる。

【0024】仮想画像44が全体画像42を横断して水平方向にスクロールされる場合、更新領域100内の新規ブロックが展開され、ディスプレイ・メモリ36内に移動される。エッジ・テーブル無しでブロックを完全に展開するには、復号されるブロックの左側の各々のブロックについてのDC差分値が加算されなければならない。それに対して、右エッジ・テーブル40を用いると、1回の減算の和だけで良いことになる。例えば、ブロックEが展開される場合、ブロックEでの絶対DC値はブロックEについてのMCU内の差分DC値から見つけられ、ブロックBについてのDC絶対値は右エッジ・テーブル40内に保存される。各々の新しい桁が仮想画像44内に導入されるに従って、エッジ・テーブルはその新しい桁についての絶対DC値で更新される。例えば、ブロックBについての絶対DC値を保持しているエントリーは、1桁左へスクロールした後ではブロックEについての絶対DC値を保持することになる。他の圧縮方式において、上部と底部のエッジ・テーブルを用いることがあり、この場合、差分値は復号されるブロックの上下のブロックを参照する。得られた仮想画像104と得られたエッジ・テーブルL'及びR'が図示してある。

【0025】2つのエッジ・テーブル38、40は、厳密には必要ではないが、2つのテーブルが使用された場合には、スクロール中に必要とされる計算が大きな仮想画像の場合に減少する。例えば、右エッジ・テーブル40が存在していない場合、ブロックEについての絶対DC値は左エッジ・テーブル38から決定されることになり、仮想画像44の最上行の各ブロックについての差分DC値を加算することになる。一つのエッジ・テーブルを用いる場合でも、エッジ・テーブルを有する仮想画像の辺縁と全体画像の辺縁の間のブロック数により必要とされる加算回数が減少する。エッジ値計算部26は仮想画像44内のブロックを含めたブロックについての絶対DC値を決定するので、これらの値はMCUを復号する

ためにこの値を使用するブロック復号部24に提供することができる。

【0026】仮想画像44がスクロールされる場合、スクローラ30はエッジ・テーブル38、40内のDC値を用いて仮想画像内にスクロールされるMCUについてのオフセットを提供する。例えば、全体画像の205桁目から204桁目へ仮想画像の左辺縁が移動するように仮想画像が移動した場合、204桁目のブロックと新しい仮想画像内部のブロックについてのMCUも復号されてディスプレイ・メモリ36内に配置される必要がある。203回の減算を実行して204桁目のブロックの絶対DC値を求める代りに、204桁目の各々のブロックについての絶対DC値を左エッジ・テーブル38の適切なエントリーから読出すことができる。左エッジ・テーブル38は203桁目の絶対DC値を保持するように更新される。これは204桁目の絶対DC値と204桁目のDC差分値から計算される。

【0027】同様に、仮想画像の右エッジが204桁目の場合、右エッジ・テーブル40は205桁目の絶対DC値を含む。一桁右へスクロールすると、スクローラ30はブロック符号化部24に新しい仮想画像のブロックに対応するMCUを復号させる。新しいブロックは205桁目にあり、これらのブロックについての絶対DC値は右エッジ・テーブル40内で既に利用可能である。右エッジ・テーブル40はこの後更新され、内部の各々のエントリーが205桁目のブロックの差分DC値で変更され、206桁目のブロックについてのDC値が得られる。

【0028】仮想画像が上にスクロールされる場合、エッジ・テーブル38、40は変更されないが、異なるエントリーが使用される。エッジ・テーブル38、40は全体画像42の長さ延在しているので、仮想画像はエッジ・テーブル38、40内の何れかのエントリーを変更する必要なしに全体画像の上部から底部までスクロールすることができる。

【0029】画像エディタ28は、ディスプレイ・メモリ36の内容を観察しつつ画像を編集し得るような対話型装置であり、又は画像エディタ28は自動処理のことが有り得る。ただし、利用者がディスプレイ・メモリ36の内容を観察しつつ又は観察せずに、一つの用途では、利用者による命令入力に回答して複写機内の画像を変更している。編集が保存される場合、画像更新部32は編集ブロックを編集MCUに再圧縮し、編集MCUを編集画像データ領域20内の利用可能な位置に保存し、そのブロックに関連するポインタ・アレイ34内のポインタを更新して新規に保存した編集MCUを示すようにさせる。画像更新部32は、また、新規保存したMCUがどのブロックを置換するかを表わす編集MCUに伴う参照フィールドも保存する。この方法だと、画像エディタ28が「取消し」操作を起動した場合、画像更新部3

2は単にポインタを変更して新規に保存したMCUによって参照されるMCUを示させ、効率的に直前の編集を取消することができる。画像更新部32及び画像エディタ28は、また一つ以上のブロックについて同時に操作する能力を有する。画像更新部32は各々の編集MCUについてのポインタ・アレイ34を更新するので、ポインタ・アレイ34は最新の状態にあり、新規の圧縮画像が圧縮画像メモリ12内に読込まれるまで、圧縮画像メモリ12はポインタ生成部22により走査される必要がない。

【0030】図5はポインタ・アレイ34からのポインタ35を示す。解説した好適実施例において、ポインタは32ビットであり、編集フラグ50を保持するための最上位ビットと残り31ビットがMCUアドレスを構成する。仮想編集はツールを用いてディスプレイ・メモリ36内の画像データを変更することで実現される。ブロック内の画素が編集されると、そのブロックのポインタ内の編集フラグが設定され、編集ブロックについての絶対DC値に編集が影響し得ることから、編集ブロックの右側のブロックのポインタ内の編集フラグも設定される。これは、右側のブロックでの差分DC値が左側のブロックでの絶対DC値で変化するため、そのブロックも編集されたことになるためである。編集作業が完了すると、ポインタ・アレイ34は設定した編集フラグが走査される。各々のフラグが立てられた領域についてMCUが圧縮画像12から復号され、フラグの立てられた領域のブロックの各列について左側に見えるMCUから開始して、各々の編集ブロックの左側のブロックでのDC値を決定する。編集したMCUが左側又は右側のエッジに到達した場合、DCエッジ・テーブルは新しい値で更新される。

【0031】図6に図示したように、新しい編集データは編集画像データ領域20内の原本の画像データ領域18に追加される。よって、ポインタ・アレイ34は単一の線形メモリとして圧縮画像メモリ12をアドレスすることができる。編集MCUが作成されれば、図6に示した編集ヘッダ・フィールドが初期化され、編集MCUが領域20内に保存され、ポインタ・アレイ34内のポインタが初期化されて新規保存した編集MCU内における編集ヘッダの始点を示す。

【0032】図6において、各フィールドは、MRKR=JPEG APPn marker for scan and block id purposes (走査及びブロックID用のJPEG APPのマーカ)

HLEN=Length of block + in bytes to conform to JPEG standard (JPEG規格に適合させるためのブロック+2のバイト長)

EID =Edit id number for unique id signature when searching (検索時の独自IDシグネチャ用編集ID番号)

PBP =Previous bit position of previously edited same mcu (直前に編集した同一MCUの直前ビット位置)

NBP =Next bit position of subsequently edited same mcu (続けて編集した同一MCUの次のビット位置)

CBB =Number of bits comprising following encoded data (これ以下に符号化されているデータを含めたビット数)

10 CBD =Recompressed edited mcu image data (再圧縮された編集MCUの画像データ)である。

【0033】即ち、MRKR及びHLENフィールドは、編集MCUがJPEG規格に適合するように提供されている。PBPフィールドはMCUの直前の版を示す。直前の版は第1の編集ブロックのMCUについて領域18内にあるが、編集したMCUがまた編集された場合には、第2の編集MCUのPBPフィールドは第1の編集MCUを示すことになる。第1の編集MCUのNBPフィールドは第2の編集MCUへ前向きに示すように設定されることになるので、リンクされたリストを構成する。CBBフィールドは後続する画像データのビット数を示し、所望するヘッダが見つかるまで一つのヘッダから別のヘッダへ飛び越すことにより画像ファイルが走査できるようになる。CBDフィールドは画像データそれ自体を含む。さらに、CBDフィールド内のデータはエントロピー符号化されているので、ビット数はMCU毎に異なる。これらのフィールドにより編集画像ファイルが新規画像ファイルに順次読込まれるようにでき、編集MCUが新規画像ファイルの原本データ領域内に含まれているような画像ファイル内に編集領域のMCUを再配置し、さらに、新規画像ファイルの編集領域が空になる。

【0034】上記の説明は、解説のためでありこれに制限するものではない。本発明の多くの変化が、本開示の参照により、当業者には明らかとなろう。単なる例として、画像処理部14の別個に識別されたユニットがコンピュータ上で動作する一つ又はそれ以上のソフトウェアプログラムとして実現され得るものであり、又はこのユニットの機能が統合され又はさらに細分されることが有り得る。本発明の範囲は、従って、上述の説明を基準とすることなく決定されるべきもので、添付の請求の範囲を参照しこれと等価の範囲に従って決定されるべきものである。

【0035】

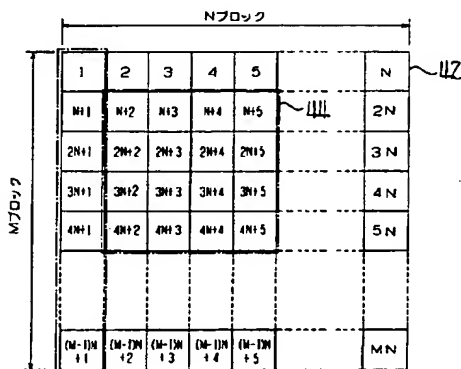
【発明の効果】本発明は、上述したように構成したので、画素レベルで圧縮された形状のままの画像の操作が可能となり、特に、典型的な画像圧縮方式を使用する画像圧縮システムにおいて、ポインタ・アレイが圧縮画像ファイル中の多くのMCUの各々を指すために提供さ

れ、画像のブロック全体からブロックの部分集合を仮想画像として選択することができる。ここに、仮想画像が編集され、各々の編集ブロックが編集ブロック内に圧縮され、原本の最小符号化単位へのポインタが編集された最小符号化単位を指すように変更され、これが編集画像データ領域内に保存されるので、この方法によれば、ポインタ・アレイが取消操作を実行するように変更させることが可能となる。

【0036】また、本発明によれば、全ての最小符号化単位が復号されていないような最小符号化単位内の差分符号化した値を取扱うための手段が提供されている。即ち、エッジ・テーブルは値を保持するために用意されており、ここで、各々の値は仮想画像の辺縁上のブロックについての差分値と結合された場合、仮想画像の辺縁より遠方のブロックを参照することなくそのブロックについての絶対値を提供するので、エッジ・テーブル内のエントリは、ブロックの完全な展開なしに圧縮された最小符号化単位から決定されるものとなる。さらなるテーブルのために占有される記憶空間より速度が有意義であるような場合に、差分力絶対値を計算する速度を向上させるために、一つ以上のエッジ・テーブルが用意されており、ここに、絶対値計算は平均してエッジ・テーブルが多いほど高速であるが、これは計算速度がエッジ・テーブルを有する最も近い辺縁までのブロックからの距離に比例するためである。

【0037】本発明によれば、また、全ての最小符号化単位を並べ替え原本の画像データ領域からの最小符号化単位を編集した画像データ領域からの最小符号化単位で置換するために圧縮された編集画像の再インデックスを取るための手段も提供されている。例えば、J P E G規格に従って圧縮された画像データで動作するが、本発明によれば、ブロックが可変容量最小符号化単位に圧縮さ

【図2】



れるような、又はデータが差分値で符号化されることにより最小符号化単位を完全に復号するためには、別の最小符号化単位を参照しなければならないようなあらゆる画像圧縮方式について応用が可能となる。

【0038】さらに、本発明による仮想編集装置によれば、圧縮画像は一つのメモリ領域内に読込まれ、また、編集すべき選択されたブロックが一つのメモリ領域から展開されて編集装置に渡され、編集され、圧縮回路によって再圧縮された後、このように再圧縮されて編集されたブロック発議に編集されたブロックメモリ内に配置されるので、ポインタ・アレイは、アレイ内のポインタが編集後の画像の各々のブロックを指し示すように、編集されたブロック・メモリの一つのメモリ領域内の何れかのブロックを示すために提供されるものとなる。

【図面の簡単な説明】

【図1】本発明の一実施例を示す仮想編集システムのブロック図である。

【図2】ディスプレイ・メモリのメモリ・マップと2次元アレイ中の圧縮画像を示す模式図である。

【図3】各々が線形アレイ内の最小符号化単位とポインタ・アレイを含む画像ファイルを示す模式図である。

【図4】仮想画像と対応するエッジ・テーブルを示し、エッジ・テーブルの使用を図示した模式図ある。

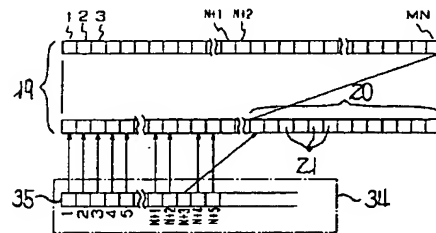
【図5】ポインタ・アレイからのポインタを示す模式図である。

【図6】編集した最小符号化単位の編集ヘッダのフィールドを示す模式図である。

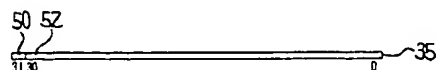
【符号の説明】

- 1 2 圧縮データメモリ
- 1 6 仮想画像メモリ、編集データメモリ
- 3 4 ポインタ・アレイ
- 4 4 仮想画像

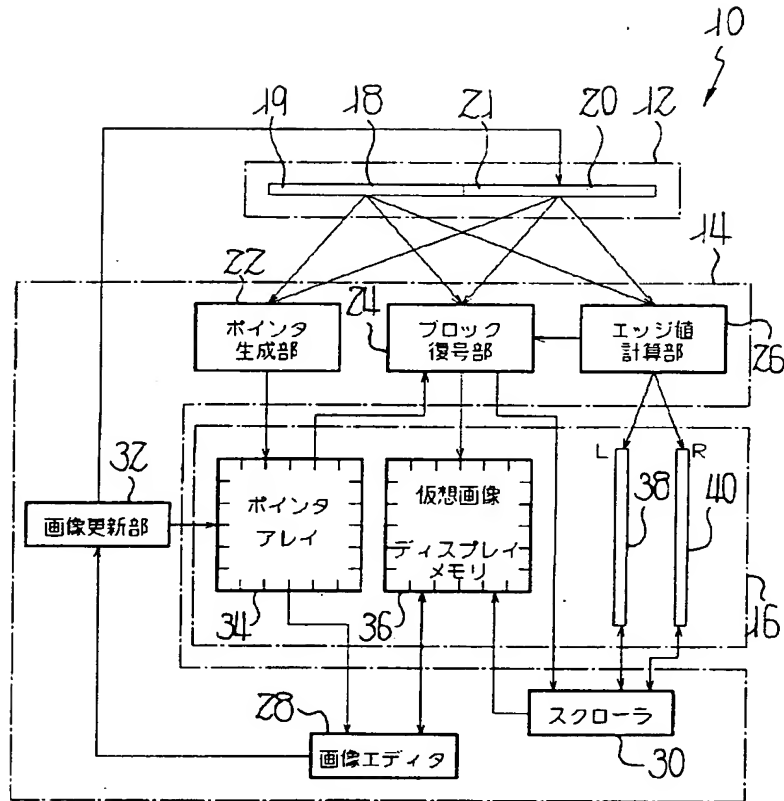
【図3】



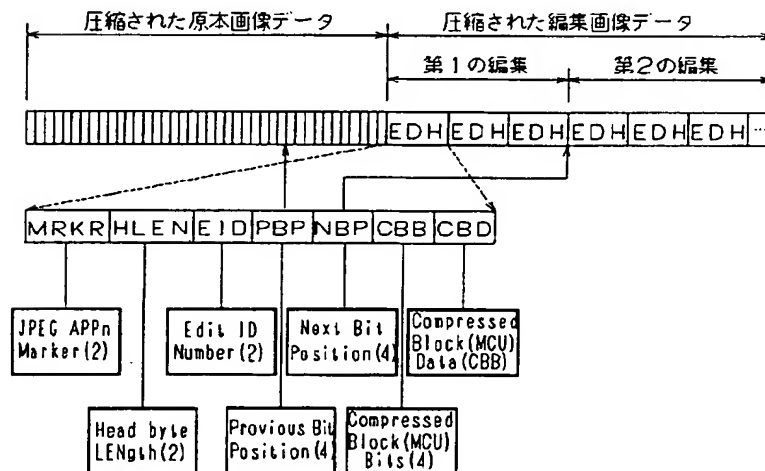
【図5】



【図1】



【図6】



【図4】

